REMARKS/ARGUMENTS

Reconsideration of this application is respectfully requested.

Initially, it is noted that the office action summary page erroneously includes a check mark in both boxes 2a and 2b. Since this is a first office action after the filing of an RCE, box 2a should not have been checked, and it is assumed that the Examiner will agree that the outstanding office action was a "non-final" action.

The Examiner is thanked for providing a "response to arguments section" bridging pages 2-4 of the office action.

The Examiner disagrees with applicants' earlier arguments that Coffman does not teach storage and dynamic updating of input/output type data when certain properties change, output prompts are sent or input responses are received. To support this disagreement, the Examiner paraphrases bits and pieces from Coffman at paragraphs [0060], [0061] and [0053]. In these paraphrased renditions, the Examiner notes that the Coffman dialogue manager and arbitrator façade (DMAF – actually, an application program interface; see paragraph [0041]) provides a mechanism for conveying application properties to the conversational virtual machine (CVM) through the dialogue manager and arbitrator (DMA). In effect, the algorithm string utilized as the API includes enough information to enable a dialogue using identified input/output resources.

From this, the Examiner concludes that storing input and output type data is implied – and asserts that such "can be dynamically updated following every response, since resources application [sic: resource applications] are updated based on user input and output response."

It is respectfully submitted that the Examiner is in error. Merely because Coffman uses a program interface which somehow manages to convey enough information to cause a dialogue on designated input/output resources does not imply that there is any stored I/O type data which is (rather than "can be") dynamically updated following every response. Furthermore, the sections cited by the Examiner from Coffman do not teach or suggest that resource applications are updated based on user input and output responses following every response. Indeed, instead of the paragraphs cited by the Examiner, the Coffman description of I/O management begins at paragraph [0147] and is associated with Fig. 8. There, it will be noted that a module known as the CAF (conversational application framework) controls engine access and arbitration. It will be noted that Coffman teaches, in paragraphs [0147] *et seq.*, that the use of a CAF (whether located in the CVM or otherwise) does not teach the storing of any type of dynamically updated I/O output type data adapted to reflect a user's preferences. Instead, it appears that Coffman in some manner "arbitrates" ambiguous user input modalities and or merely accepts

whatever input modality happens to be chosen by the user. There does not appear to be any storage of user preferences as to I/O type modalities, or any active steering of a user towards any particular modality (whether based on a detected prior user preference or otherwise).

The Examiner refers to a mention in Coffman of "user preferences" occurring at paragraph [0175]. At paragraph [0175], Coffman states "the user may interact with the different applications offered by the portal based on, e.g.,...user preferences..." This section of Coffman clearly teaches the possibility that a user may have preferences, but does not teach the apparatus of claim 1 comprising means for establishing a user preference value from stored data indicative of utilization made by a user. That is, this passage is consistent with the Coffman description of I/O management at paragraphs [0147]-[0172]. Merely because Coffman somehow permits I/O in multiple modalities does not teach, imply or suggest that Coffman determines and stores (and updates) user preferences for I/O modalities.

Moreover, the applicants' claimed storage of input and output type data indicative of the utilization of ports by a user is not found in Coffman.

Coffman does, of course, describe a dialogue manager and arbitrator (DMA) that seeks to identify the mode of user input and to match that input to a suitable application –

i.e., an application capable of handling that mode. The present inventors have gone at least one step further by establishing a stored measure of the mode (or modes) of interaction preferred by the user. The present inventors have, as a result, provided a dialogue apparatus that is capable of selecting from multiple inputs or outputs, the input or output that is able to convey information in a mode most easily processed by the user – something that is not offered by Coffman.

For example, if it is found that motor-input (i.e., keyboard) is preferred over audio-input (even though both are supported), the unused modality could be allocated to a 'supportive' role. That is, rather than using modality-independent wordings (such as "what is your surname?") for audio-output, the alternative "please *enter* your surname" could be used. The audio-output prompt directs the caller to use their modality of choice (i.e., the keyboard). The system, as a result, has become supportive of the caller's preferred modality. This is an area not addressed by Coffman which teaches no means to achieve the enhanced user interface provided by the apparatus and methods claimed.

The rejection of claims 1-11, 13-33 and 35-46 under 35 U.S.C. §102 as allegedly anticipated by Coffman '174 is respectfully traversed.

It is respectfully noted that the now pending claims are claims 1-5, 7-11, 13-27, 29-33 and 35-46. Presumably, the Examiner intended to cite these claims as the list of rejected claims.

For reasons already of record and, to some extent, recounted above, Coffman does not teach (or suggest) every element of any rejected claim – a prerequisite for alleged anticipation.

For example, as previously noted, Coffman does not teach a store which stores I/O type data indicative of one or more properties of the I/O ports and the input responses and output prompts communicated therethrough – wherein such type data is updated when (a) any of said one or more properties change, and/or (b) output prompts are sent and/or (c) input responses are received.

Nor does Coffman teach any means for establishing from any recorded data, for each of the I/O ports, a user preference value. Independent apparatus claim 2 and independent method claims 23 and 24 include analogous requirements that are in no way taught (or suggested) by Coffman. The dependent claims therefore also include such limitations.

Accordingly, it is note believed necessary at this time to detail additional deficiencies of Coffman with respect to other aspects of the rejected claims. Suffice it to

note that, as a matter of law, it is impossible for any reference to anticipate a claim unless it teaches each and every feature of that claim.

A formal notice of allowance is earnestly solicited.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: Larry S. Nixon

Larry S. Nixon
Reg. No. 25,640

LSN:lef

901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100